Biometric System Based on Neural Network for Efficient Management of Academic Attendance

Alexandru Borta
Faculty of Computer Science,
"Alexandru Ioan Cuza" University
General Berthelot, 16,
Iasi, Romania
alexandru.borta@info.uaic.ro

Adrian Iftene
Faculty of Computer Science,
"Alexandru Ioan Cuza" University
General Berthelot, 16,
Iasi, Romania
adiftene@info.uaic.ro

ABSTRACT
The system developed in this paper aimed to design a student attendance system that could effectively manage the academic presence in Romanian universities. In the proposed system, the identification of students take place through fingerprints that provide a high level of uniqueness due to the fact that there cannot exist two individuals with the same fingerprint. The system consists of the following five key components: (1) a MySQL data base for storing all data, (2) Xamarin.Mac desktop application used to register students and teachers in the database, (3) a NodeJS web application for live viewing of attendance records, (4) a Nodejs server responsible for the fingerprint matching algorithm, and (5) two Arduino prototypes with fingerprint modules used to record students and attendance in the system.

Author Keywords
Biometric identification; fingerprint matching; neural network.

ACM Classification Keywords
H.5.2. Information interfaces and presentation (e.g., HCI): User Interfaces. H.3.2. Information Storage and Retrieval: Information Storage.

General Terms
Human Factors; Design; Measurement.

DOI: 10.37789/rochi.2023.1.1.21

INTRODUCTION

Currently, in the majority of universities worldwide, as well as in universities in Romania, the management of student attendance is done through traditional methods. Despite the mandatory nature of most activities, the institution's leadership does not enforce a predefined protocol for tracking attendance. As a result, this responsibility remains at the discretion of each individual teacher. Typically, at the beginning of the teaching activity, if the lecturer does not have a list, one is prepared with all the students present in that activity, which is usually circulated around the class while the lecturer conducts the activity. At the end of the academic year, these lists are used to track whether students have accumulated the minimum required number of attendances to be eligible for the subject's promotion exam. The fundamental principle of universities is centered around promoting justice and intellectual equity. However, like any other traditional system based on human factors [1], this system has numerous disadvantages and potential errors that may arise from its use. For instance, transmitting the attendance sheet in a large classroom can be quite challenging, considering that each student may occupy a seat at a considerable distance from another student. In addition to the physical transmission of the sheet, manual signing by each student can prevent the teacher from receiving full attention from the students and can also distract the students from focusing on the material being taught. Moreover, as the attendance sheet is passed from student to student, some may accidentally or intentionally sign another student's name, resulting in inaccurate records. On average, with a class size of 40 students, this process would consume approximately 10-15 minutes for creating the list and around 5 minutes for the teacher to verify it. On average, the curriculum for an academic year consists of 14 academic weeks, with a minimum of 2 hours per week per subject. This would mean a total of 14 academic activities per subject. If each activity lasts 2 hours, and the time lost for creating and verifying the attendance sheet is an average of 17 minutes, it would result in approximately 4 hours lost during an academic year, equivalent to 2 academic activities. The main purpose of this project is to improve the current paper-based system and other existing digital systems by creating a better and more affordable system that any university can afford. The proposed system will address the following listed problems: (1) *Lack of backup copies for attendance records* in case the lecturer accidentally loses the attendance sheet. (2) *Erroneous attendance records* were created due to friendly signing (where attending classmates sign for absent students). (3) *The complexity of analyzing physical format records*. (4) *Tracking student performance based on attendance factor*.

Managing student attendance records in a university is a tiresome task. In order to automate all attendance-related activities, the project objectives are divided into two categories: software development objectives and hardware development objectives. The *hardware* part of this system has the following objectives: (1) *Creation of a student registration prototype* used for obtaining digital fingerprints during the student registration process in the database via a USB connection that will not exceed the cost of ten euros. (2) *Creation of a prototype for recording* attendance obtained through fingerprint matching and transmitting it to

the server via a wireless connection that will not exceed the cost of ten euros. This type of prototype is intended to be installed at the entrance of all rooms where academic activities take place. The *software* part of this system has the following objectives: (1) *Creation of a structured database* for storing all obtained data (*student personal data, faculty personal data, schedule, subjects, attendance records*). (2) *Creation of a server* and implementing a REST API with a minimum of two endpoints (for enrollment and attendance registration). (3) *Development of a desktop application* that allows student enrollment using digital fingerprints, as well as faculty enrollment. (4) *Creation of a web application* that enables live viewing of attendance records with access through username and password (account creation will be available only to faculty members). The paper is structured into three important parts: theoretical foundation, creation of fingerprint reading prototypes, and implementation of the student attendance management system. The starting point of the study presents attendance management systems in universities outside Romania and their requirements. The analysis covers the current evolution of technologies used, as well as the advantages and disadvantages of their implementation within the structure of an attendance management system, by examining several similar systems other than the traditional one. Each subsequent section refers to a component of the newly created system. Digital fingerprints represent the core technology of the system and the fundamental property of the created prototypes [2].

LITERATURE REVIEW

Due to the continuous development of new technologies and the large number of research centers in this field, most human-based systems are being replaced with computerized systems [3, 4]. Currently, there are already several types of automated attendance management systems that have been introduced in some schools, colleges, and universities.

RFID-based Attendance Management System

One approach to the attendance management problem is the use of a RFID-based system. Radio Frequency Identification (RFID) is a technology that uses radio waves to transfer data from an electronic tag, also known as a RFID tag (attached to an object), to a reader for identification or object tracking purposes [5]. RFID technology is a mature technology widely used by various organizations as part of their automation systems. In [6], the authors describe such a system. The article aims to replace the traditional system with a RFID-based system. According to the authors, this system would address some of the issues present in the traditional attendance management system and can be used in both educational institutions and private companies. This system consists of two modules - the RFID module and the reading module. The main idea is that each student must have a valid RFID card that they can place near the reading module located at the entrance of each study room. The reading module will automatically detect the student's identity and record their attendance. The given management system has several key advantages. First, this type of system can be considered a fully automated system that requires minimal human involvement in the attendance registration process. This is because the student only needs to place the RFID card near the reading module at the entrance of the study room. In comparison to the traditional system where a physical attendance sheet circulates among students, this process is rendered unnecessary, significantly reducing the time required. Another important advantage is that the information provided by the system will be much more accurate than manually recorded records on paper, as they are created by the computer. In addition to the mentioned advantages, a RFID-based identification system has a few disadvantages. Due to the fact that attendance records are based on RFID cards, students will need to carry their cards with them at all times. If a student loses their card, they will have to go to the office, report the loss, pay for the lost card, and wait for a new card to be issued. As we know, a student participates in multiple courses, which means that in case of losing the card, the student will have to inform the lecturer. The lecturer, in turn, will have to notify the office that the student was present at the academic activity because the system does not provide a way to correct or edit attendance records. Furthermore, this system can prevent erroneous attendance registration, but not with 100% accuracy. There is a possibility that a student may lend their RFID card to another student, resulting in an erroneous attendance record in the system.

Barcode Sensor-based Attendance Management System

Another approach to the attendance management problem is the use of a system based on barcode scanners. Barcode identification is one of the current automatic identification technologies even today. A barcode is an optical representation that can be read by a barcode scanner, where each code identifies an object [7]. In [8], the authors present such a system. The article aims to replace the traditional system with a barcode scanner-based system. As mentioned in the article, a RFID-based system or a biometric-based system could be quite costly to implement in an educational institution as they require the acquisition of many hardware components. Compared to RFID-based and biometric-based systems, barcode technology is less expensive. In this system, each student will have a student card with a unique barcode that will be scanned at the beginning of each academic activity. The presence of each student will be automatically recorded in the system when the professor scans the student's barcode. The given management system has several key advantages. Firstly, this type of system can be considered almost fully automated as it requires human involvement only for scanning the barcodes. Another considerable advantage is the cost of this system. Due to the widespread use of this system, there is a wide range of scanners available on the market at an advantageous price. In addition to the price advantage, the system described in the article automatically generates a warning letter if the student does not have the minimum required attendance to pass the

academic activity. In addition to the aforementioned advantages, a barcode sensor-based system also has a few disadvantages. Since the records are based on the barcodes on students' cards, they will have to carry their cards with them at all times. If a student loses their card, they will have to go to the office, report the loss, pay for the lost card, and wait for a new card to be issued. Another disadvantage is that the professor will spend approximately ten minutes of the teaching activity to scan the barcodes of all the students, and this is one of the main issues that an automated system should solve. From the above, we can identify several limitations or weaknesses of the systems described.

Overcoming limitations

The proposed solution in this paper will attempt to improve these weaknesses and eliminate the limitations present in similar systems. One of the main limitations of the systems described above is the unique identifier of the students, which can be lost. In the proposed system, the student will be registered based on their fingerprints, which means that the student cannot lose their identifier. Moreover, fingerprints provide a high level of uniqueness because no two individuals have the same fingerprint [9]. A weakness present in both solutions is the occurrence of erroneous records. RFID cards, as well as student cards with barcodes, can be borrowed from one student to another. There is a possibility that some students may not want to participate in an academic activity but record their attendance through other students. This would directly affect the accuracy of the records, resulting in erroneous entries. Since the proposed system is based on fingerprints, recording attendance without the physical presence of the student is impossible

ANALYSIS AND THEORETICAL FOUNDATION

Biometric Identification

Biometric identification systems are widely used for unique identification of individuals, primarily for monitoring and identification purposes. A biometric system [10] is essentially a pattern recognition system that performs personal identification by establishing the authenticity of a specific physiological or behavioral characteristic possessed by the user. Logically, a biometric system can be divided into the *enrollment module* and the *identification module*. In the *enrollment phase*, an individual's biometric feature is first scanned by a biometric sensor to obtain a digital representation of the feature. To facilitate matching and reduce storage requirements, the digital representation is further processed by an extractor to generate a compact yet expressive representation called a "template". Depending on the application, the template can be stored in an external database or in the memory of the biometric sensor if it allows it. The utilization of biometrics in identity management and access control [11] establishes its significance in ensuring secure operations. Hence, incorporating biometrics into the student attendance management system is a reliable and secure approach.

Generic Biometric System

A fingerprint represents a pattern of ridges and valleys on the surface of a finger [1]. The endpoints and intersections of the ridges are also known as minutiae. It is widely accepted that the minutiae pattern of each finger is unique and does not change over a person's lifetime [12]. When human fingerprint experts determine whether two fingerprints are from the same finger, the degree of matching between the two minutiae patterns is one of the most important factors. Due to the uniqueness of minutiae patterns, the minutiae-based matching method is currently the most studied method.

Hardware Technologies Involved

NodeMCU DEVKIT 1.0[1] is a development kit based on ESP8266 that integrates GPIO, PWM, IIC, 1-Wire, and ADC all on a single board. It has a NodeJS-style API network, which allows developers to write code in a NodeJS style that runs on a 5mm* MCU. There are two main reasons for choosing this DEVKIT: its low price (approximately 3 Euros) and the integrated Wi-Fi module.

Fingerprint Sensor FPM10A is an Arduino-compatible fingerprint module. With its high-speed DSP processor, it could also work with other serial devices such as MSP4230, 51, AVR, PIC, STM32, ARM, and FPGA. This fingerprint module has the capability to operate independently without any host computer or PC software. FPM10A can store up to 1,000 fingerprints and supports fingerprint scanning, intelligent image processing, fingerprint comparison, and fingerprint searching mode. It also has a high sensitivity for recognizing wet and dry fingerprints [13].

Monochrome LCD Nokia 5110[2] with 84×84 pixels was used in old phones like Nokia 5110/3310. They have a diagonal size of about 1.5" but are highly readable and equipped with a white backlight. Since they consist of 84×84 individual pixels, they can be used for graphics, text, or bitmaps. Their price does not exceed 4 Euros, and they are very easy to use, requiring only a few digital I/O pins. To operate the display, 3 to 5 digital output pins are needed (depending on the need to manually control the selection and reset lines of the chip). The display driver is a PCD8544 chip and operates at 3.3V. The logic levels must be 3V to prevent damage, but thanks to the built-in level shifter, the LCD can be connected to 5V logic, such as Arduino boards.

IMPLEMENTATION

System Architecture

The implemented system architecture is presented in Figure 1. It consists of a client-server structure composed of

[1] https://www.nodemcu.com/index_en.html

[2] https://www.digikey.com/catalog/es/partgroup/nokia-5110-3310-monochrome-lcd/63437

elements that request services and a service provider (the NodeJS server). The main components of the system are: (1) a *desktop application* and a *fingerprint sensor prototype*, also known as an *enrollment prototype*, used for enrolling students and professors into the database through the server, (2) a fingerprint sensor prototype, also known as an *attendance prototype*, which will be installed at all entrances of the teaching activity rooms for recording attendance, (3) a *web application* that allows registration/authentication of teachers for live viewing of attendance, a NodeJS server, and (4) a *MYSQL database*. The server will be responsible for providing data for the web application, enrolling students/professors into the database, recording attendance, and matching fingerprints for them.

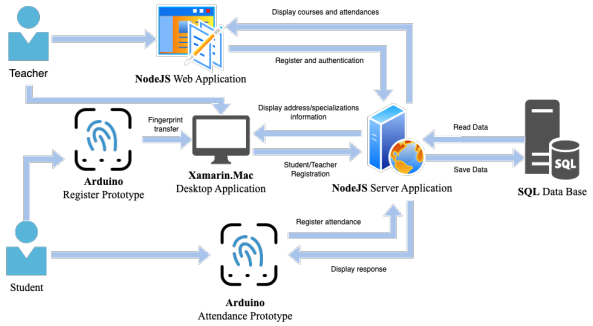

Figure 1. System Architecture.

Hardware Design

The working principle of the presented attendance system is divided into two parts and explained separately due to the two types of prototypes. These prototypes need to ensure the registration of a fingerprint in an existing database (registration prototype) and save a student's attendance record (attendance prototype). In order to establish the necessary connections between the chosen microprocessor and the other modules, there are several requirements, that need to be fulfilled. Figure 2 shows the pinout of the NodeMCU v.1 board and their hardware names[3].

Desktop application

Another objective of this work is to create a desktop application that will allow enrolling students and teachers using fingerprint recognition. One of the most important functionalities of this application is establishing a connection and communication with the registration prototype. As mentioned earlier, attendance templates will be sent and stored in the database as one-dimensional tables of decimal values. The purpose of this application is to record these templates (13 in total for each student) along with other information such as Name, Surname, Address, Academic Group, and Specialization in the database. The desktop application allows the addition of two types of records: Student records and Teacher records.

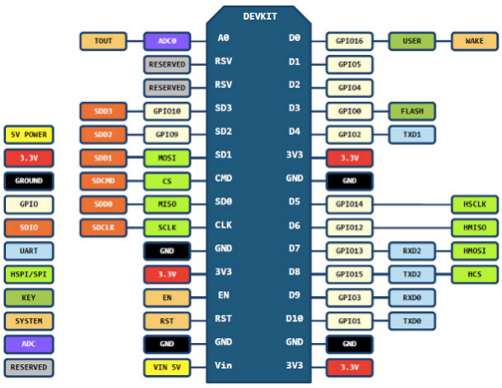

Figure 2. NodeMCU v.1 Pin Map.

Registration Prototype

The architecture description of the prototype is presented in Figure 3. Communication between the registration prototype and other system components will take place through a serial port, which will also be used to power the microcontroller. To send a fingerprint through the serial port, the fingerprint image needs to be transformed into a data type supported by the serial port. The "Get Template" method retrieves the fingerprint from the microprocessor's flash memory and converts it into a decimal value array of length 256. The obtained data array is divided into 16 one-dimensional arrays to fit into a serial port line (32 bits - maximum length of a line / 2 bits - one hexadecimal value is represented by 2 bits).

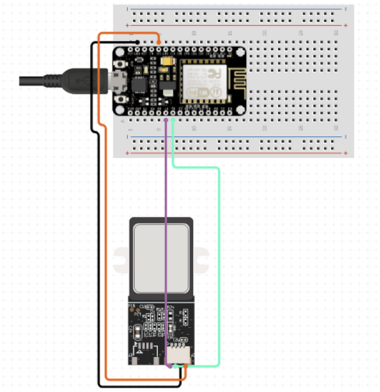

Figure 3. Circuit diagram of the registration prototype[4].

Attendance Prototype

The architecture description of the prototype is presented in Figure 4. The purpose of the attendance prototype is to obtain the fingerprint template from the sensor and send it to the server through a POST HTTPS request, where the matching process takes place. The server records the attendance and sends back a response - the scanned student's name if there is a match or an error message if there is no match. The connection between the prototype and the server will be an

[3] https://www.nodemcu.com/index_en.html

[4] Made with: https://www.circuito.io/

internet connection. The attendance prototype consists of three hardware modules: the NodeMCU microcontroller, an OLED display, and the fingerprint sensor. The port numbers are chosen by the developer, meaning that any connection that meets the configuration and pin description of the chosen microcontroller will work. Communication between the attendance prototype and other system components will take place through the HTTP protocol. The serial port will be used to power the microcontroller. To use the presented system, a permanent and secure network is required. The access point credentials, as well as the connection information to the NodeJS server, are established by the developer. If any changes occur (such as changing the server's IP or access point credentials), reconfiguring the prototype is necessary.

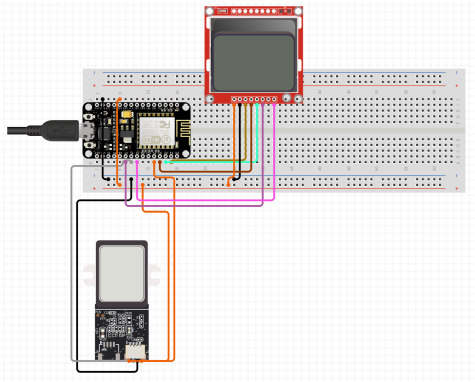

Figure 4. Circuit diagram of the attendance prototype[5].

Database

One of the main objectives of this work is to create a structured database used for storing all the obtained data (*student personal data, teacher personal data, schedule, subjects, attendance records*).

Teacher table contains all the information about each teaching staff member. This table will also be used in the web application for the authentication/registration process. When creating a *Teacher* record, the *Password* field is initialized with null, meaning that the professor is registered in the university's database but does not have an account for the web application. The *Student* table contains all the information about each student. The field *id. academic_group* establishes a connection between the student and their enrolled academic group. This field is used to determine, during the attendance registration process, whether the student is required to participate in a specific academic activity or not. The *Address, Region*, and *City* tables contain information about the location where the student/professor resides. When registering a student/professor in the database, an entry will initially be created in the Address table, and only after that, a student/ professor record will be created with the ID of the recently created address. The *Academic Group* table contains information about all the groups existing in the university where this system will be implemented. A group represents a combination of specialization and academic year, e.g., Computer Science 4. The *Specialization* table contains information about all the specializations existing in the university where this system will be implemented. The *GroupSubjects* table serves as a link between the Subject and Group tables to establish a many-to-many relationship between records. Multiple groups can have the same subject, and conversely, a group can have multiple subjects. The *StudentsFingerprints* table contains biometric information for all registered students in the system. Each student will have 13 biometric records saved in this table. The *ActivityRoom* table contains information about each room where an academic activity can take place, along with the attached prototype presence identifier. The *Subject* table contains information about each academic activity within the university. The *Attendance* table contains information about each student's attendance in an academic activity. These records represent the primary information on which the web application will be based. To add an attendance record to the database, several important validations are required. To address this, a stored procedure has been implemented to take care of these aspects.

Desktop application

Another objective of this work is to create a desktop application that will allow enrolling students and teachers using fingerprint recognition. One of the most important functionalities of this application is establishing a connection and communication with the registration prototype. As mentioned earlier, attendance templates will be sent and stored in the database as one-dimensional tables of decimal values. The purpose of this application is to record these templates (13 in total for each student) along with other information such as *Name, Surname, Address, Academic Group,* and *Specialization* in the database. The desktop application allows the addition of two types of records: *Student* records and *Teacher* records. The retrieval of fingerprint templates takes place in the "Retrieve biometric data" block. Once the registration prototype is connected to the computer unit, a serial connection is established, through which the fingerprint templates will be transmitted. The data of a student's registration set transmitted through the serial port follow a set of well-defined rules.

- A registration set starts with the character "Put".
- Each fingerprint template sent will end with "Done".
- Any error that occurs at the registration prototype level (unclear image, incomplete image) will end with "Error".
- The end of a registration set will be marked by the "#" character.

[5] Made with: https://www.circuito.io/

Each biometric data registration loop will take place on a separate thread to avoid blocking the UI thread.

NodeJS Server

For security and performance reasons, the communication between the entire system and the MySQL database must take place from a single point. This is the main purpose of this server. The matching of fingerprints at the moment of recording an attendance must take place on the server side. Currently, there is no open-source fingerprint-matching algorithm available that could be used in the proposed solution. An algorithm based on minutiae features would have been an ideal solution, but the current fingerprint module sensor does not provide fingerprints in an image format. A solution that partially meets the requirements of a matching algorithm is the implementation of a neural network, where fingerprint templates would serve as inputs and the IDs of registered students in the database would serve as outputs. This dataset is stored in the *StudentsFingerprints* table. Once the learning process is completed, the server will save the trained network as a JSON file. This JSON file will be used for subsequent server launches. In case this file is missing, the server will initiate a new learning process using the dataset.

Web Application

Any management system should provide a graphical application for visualizing the data flow. In the case of this system, the GUI is a NodeJS web application. The main functionalities of this application are:

- *Registering* a professor in the given application. Registration will only be allowed for teaching staff members who belong to the university where this system is implemented. A teaching staff member belongs to the university only if their email is registered in the database. Registration in the web application will be based on email and the last 3 digits of their ID number (CNP). This approach helps avoid situations where a third party who knows a professor's email would attempt to register.
- *Authenticating* a professor in the given application.
- *Viewing the academic activities* taught by the professor.
- *Viewing the live attendance of students* for each academic activity taught by the authenticated professor.

EVALUATION

Neural network

The component responsible for fingerprint matching in this system is a neural network. A digital fingerprint has 240 features, out of which only 143 are unique (the rest being common features shared by all fingerprints). The given neural network was trained on a data set of 20 fingers (13 fingerprint templates each), totaling 260 fingerprints with 20 outputs (the IDs of the 20 registered students). Unfortunately, training the neural network with a larger dataset wasn't feasible due to the sensitivity of fingerprint data. Fingerprint information is considered highly personal and confidential, and obtaining a large dataset of fingerprints for training purposes is challenging. Currently, there is no openly available source or repository that provides a comprehensive dataset of fingerprints.

Therefore, the training of the neural network was limited to a smaller dataset, specifically 20 fingers of relatives/family with their respective fingerprint templates. Despite this limitation, efforts were made to ensure the effectiveness of the system within the available constraints. Despite the small dataset, the training process took several days. Following manual testing of the 20 fingers utilized for training, an approximate recognition rate of 65% was attained. While this percentage is relatively high, it falls short of ensuring consistently accurate results. The error results of testing are presented in Table 1.

Tested finger	Output finger	Expected finger
F1	**F5**	F1
F2	**F2**	F2
F3	**F3**	F3
F4	**F4**	F4
F5	**F5**	F5
F6	**F1**	F6
F7	**F7**	F7
F8	**F20**	F8
F9	**F9**	F9
F10	**F20**	F10
F11	**F11**	F11
F12	**F1**	F12
F13	**F13**	F13
F14	**F1**	F14
F15	**F15**	F15
F16	**F16**	F16
F17	**F6**	F17
F18	**F18**	F18
F19	**F6**	F19
F20	**F20**	F20

Table 1. Neural network manual testing results.

Overview

The system's primary goal is to decrease the time needed for attendance registration. In the current implementation - the total registration time for attendance in the system is represented as the sum of the three times (fingerprint

scanning time + processing time and response receiving time - whether it's the student's name or the error message).

$$TT = TS + TP + TR$$

The average values of these times are represented in Table 2 where:

- *TT* - total time for registering an attendance;
- *TS* - scan time of a finger;
- *TP* - fingerprint processing time;
- *TR* - result obtaining time.

TT (ms)	TS (ms)	TP (ms)	TR (ms)
6,568	2,132	3,483	953

Table 2. Average attendance registration timetable.

On average, a time of 6.5 seconds for registering attendance was achieved. The final results are represented in the Table 3 where:

- *RR* - recognition rate;
- *NSC* - the average number of scans;
- *NST* - number of students;
- *TT* - total time for registering an attendance;
- *TAPT* - total attendance process time;

and

$$NrOfFailureScans = \left(\frac{NSC \times (100\% - RR)}{100}\%\right) \times (NSC - 1)$$

$$TAPT = NST + NrOfFailureScans \times TT$$

	Perfect scenario	Realistic scenario	Actual system scenario
RR (%)	100%	80%	65%
NSC	1	2	3
NST	40	40	40
TT (s)	6.5s	6.5s	6.5s
TAPT (min)	4.3min	5.2min	7.3.min

Table 3. Final results table.

In the ideal scenario, with a fingerprint recognition rate of 100%, the total time required for attendance registration would be 4.3 minutes, representing a 71% improvement over the traditional method, which would take 15 minutes. In a more realistic scenario, where incorrect fingerprints, dirty or injured fingers, or any other factor leading to incorrect matches and requiring rescanning may occur, the total registration time would be approximately 5.2 minutes - 65% faster than the traditional method. In the current system, where the error rate for fingerprint matching is 35% and double rescanning is required compared to the other two scenarios, the total registration time would be approximately 7.3 minutes - 51.3% faster than the traditional method.

Experiments

Within the scope of this study, an overview of the outcomes from the experiments is provided, simulating the functionalities of a potential digital attendance management system implemented in a laboratory or classroom setting.

Setup Requirements

To set up the experimental system, the following was prepared:

- *An ordinary classroom with two entrances*: The room has a capacity of approximately 45 students.
- *A secure access point*: The current access point is expected to be used only by the implemented system. In our case, the user access point is an ordinary home-purpose access point.
- *One attendance prototype*: The prototype was strategically placed at the entry point to ensure easy access and minimize congestion. In the current experiment, the prototype was connected to an electricity socket through a Micro-USB cable.

To perform the simulation, the trained neural network described in the Neural Network section mentioned above was utilized. Twenty fingers from two individuals were registered. Each finger will be treated as an individual student for experimental purposes.

Experiment Results

Since attendance registration was permitted precisely at the start of the course (14:00 in our case), it was assumed that a queue of 20 students would form at the door. Following the registration of all 20 fingers, a time of 9.3 seconds was recorded for attendance registration.

The difference between the value presented in the overview of this chapter is mainly because of the access point. Because it was an access point also used by other devices, the response time was a little bit delayed. During the registration process, intentionally dirty fingerprints were employed to replicate a real-life scenario. As a result, the average number of scans per finger has grown to 4, which resulted in a 5.2 min time for the whole registration of 20 students. Compared to the value stated in the overview of this chapter it took 2.2 minutes longer which is slower by 39%, but still, 30.7% faster than the traditional method.

A solution that would lower by far the total registration time is to use multiple attendance prototypes. Using an additional prototype, or more than two, would divide the total registration time by the number of prototypes. Because the classroom used for the experiment has two entrances, it allows us to install an additional prototype that would reduce the total registration time, from 5.2 min to 2.6 min. The number of used prototypes should be decided at the

beginning of the implementation of the system for a better outcome.

CONCLUSION

The implemented solution fulfills most of the originally proposed objectives, which are:

- *Creating a structured database* used for storing all acquired data (student personal data, teacher personal data, schedule, subjects, attendance records).
- *Creating a server application* and implementing a REST API with at least two endpoints (for enrollment and attendance registration).
- *Developing a desktop application* that allows enrollment of students using fingerprint biometrics, as well as enrollment of teachers.
- *Creating a web application* that ensure live visualization of attendance records with access via username and password (account creation will be available only for teachers).
- *Building a student enrollment prototype* for obtaining fingerprint biometrics during the student registration process in the database using a USB connection that does not exceed the cost of ten euros.

One of the partially achieved objectives is the creation of a prototype for recording attendance obtained through fingerprint matching and transmitting it to the server via a wireless connection that does not exceed the cost of ten euros. Due to the limited memory size of this type of microcontroller, an external fingerprint-matching algorithm had to be implemented, as the current matching rate within the prototype was not acceptable.

One of the most important and necessary future developments is the implementation of a new fingerprint-matching algorithm. An algorithm based on minutiae features could result in a matching rate of up to 98%. Another future development would be the implementation of academic report generation for defined periods. The current system allows for live visualization of attendance records but does not generate semester or annual reports for each academic activity.

Since the desktop application was developed using the Xamarin framework, approximately 70% of the source code is cross-platform, which facilitates and simplifies the application development process for other operating systems such as iOS, Android, and Windows.

REFERENCES

1. Kumar, D. Ashok and Begum, T. Ummal Sariba. A Comparative Study on Fingerprint Matching Algorithms for EVM. *Journal of Computer Sciences and Applications 1,* 4 (2013), 55-60.
2. Pankanti, S., Prabhakar, S., and Jain, A.K. On the individuality of fingerprints. *IEEE Transactions on pattern analysis and machine intelligence 24*, 8, IEEE (2002), 1010-1025.
3. Cazuc, C. R., Iftene, A. User Recognition Based on Keystroke Dynamics. *5th Proceedings of the Conference on Mathematical Foundations of Informatics*, (2019), 343-350.
4. Gotcă, A., Iftene, A. ECG Biometrics: Experiments with SVM. *Proceedings of the Conference on Mathematical Foundations of Informatics MFOI2020* (2021), 145-160.
5. Chiagozie, O. G., and Nwaji, O. G. Radio frequency identification (RFID) based attendance system with automatic door unit. *Academic Research International 2*, 2, SAVAP International (2012), 168.
6. Arulogun, O., Olatunbosun, A., Fakolujo, A., and Olaniyi, O. RFID-Based Students Attendance Management System. *International Journal of Engineering and Scientific Research 4*, (2013).
7. Lotlikar, T., Kankapurkar, R., Parekar, A. and Mohite, A. Comparative study of Barcode, QR-code and RFID System. *International Journal of Computer Technology and Applications 4*, 5, (2013), 817.
8. Lakshmi S. K., Shinde, S., Thomas, T. and Abdugani, A. Barcode based Student Attendance System. *International Journal of Computer Applications 119*, (2015), 1-5.
9. Laughery, K. R. and Laughery, K. R. Human factors in software engineering: A review of the literature. *Journal of Systems and Software 5*, 1 Elsevier (1985), 2-14.
10. Jain, A., Hong, L., and Pankanti, S. Biometric Identification. *Communications of the ACM 43*, (2000), 90-98.
11. Wayman, J. L. Biometrics in identity management systems. *IEEE Security & Privacy 6*, 2, (2008), 30-37.
12. Adeoye, Olufemi Sunday. A survey of emerging biometric technologies. *International journal of computer applications 9*, 10 (2010), 1-5.
13. Hamidu Alkali, A., Dada, E., Mshelia, D. and Onundi, S. Design and Development of an Arduino Based Electronic Voting System. *International Refereed Journal of Engineering and Science, 8*, 1 (2019), 48-57.